Wikipedia

Passphrase

A **passphrase** is a sequence of words or other text used to control access to a computer system, program or data. A passphrase is similar to a password in usage, but is generally longer for added security. Passphrases are often used to control both access to, and operation of, cryptographic programs and systems, especially those that derive an encryption key from a passphrase. The origin of the term is by analogy with *password*. The modern concept of passphrases is believed to have been invented by Sigmund N. Porter[1] in 1982.

Security

Considering that the entropy of written English is less than 1.1 bits per character,[2] passphrases can be relatively weak. NIST has estimated that the 23-character passphrase "IamtheCapitanofthePina4" contains a 45-bit strength. The equation employed here is:[3]

> 4 bits (1st character) + 14 bits (characters 2–8) + 18 bits (characters

9–20) + 3 bits (characters 21–23) + 6 bits (bonus for upper case, lower case, and alphanumeric) = 45 bits

(This calculation does not take into account that this is a well-known quote from the operetta H.M.S. Pinafore. An MD5 hash of this passphrase can be cracked in 4 seconds using crackstation.net, indicating that the phrase is found in password cracking databases.)

Using this guideline, to achieve the 80-bit strength recommended for high security (non-military) by NIST, a passphrase would need to be 58 characters long, assuming a composition that includes uppercase and alphanumeric.

There is room for debate regarding the applicability of this equation, depending on the number of bits of entropy assigned. For example, the characters in five-letter words each contain 2.3 bits of entropy, which would mean only a 35-character passphrase is necessary to achieve 80 bit strength.[4]

If the words or components of a passphrase may be found in a language dictionary—especially one available as electronic input to a software program—

the passphrase is rendered more vulnerable to dictionary attack. This is a particular issue if the entire phrase can be found in a book of quotations or phrase compilations. However, the required effort (in time and cost) can be made impracticably high if there are enough words in the passphrase and how randomly they are chosen and ordered in the passphrase. The number of combinations which would have to be tested under sufficient conditions make a dictionary attack so difficult as to be infeasible. These are difficult conditions to meet, and selecting at least one word that cannot be found in *any* dictionary significantly increases passphrase strength.

If passphrases are chosen by humans they are usually biased by frequency of particular words in natural language. In the case of four word phrases, actual entropy rarely exceeds 30 bits. On the other hand, user-selected passwords tend to be much weaker than that and encouraging users to use even 2-word passphrases may be able to raise entropy from below 10 bits to over 20 bits.[5]

For example, the widely used cryptography standard OpenPGP requires that a user make up a passphrase that must be entered whenever decrypting or signing messages. Internet services like Hushmail provide free encrypted e-mail or file sharing services, but the security present depends almost entirely on the quality of the chosen passphrase.

Compared to passwords

Passphrases differ from passwords. A password is usually short—six to ten characters. Such passwords may be adequate for various applications (if frequently changed, if chosen using an appropriate policy, if not found in dictionaries, if sufficiently random, and/or if the system prevents online guessing, etc.) such as:

- Logging onto computer systems
- Negotiating keys in an interactive setting (e.g. using password-authenticated key agreement)
- Enabling a smart-card or PIN for an ATM card (e.g. where the password data (hopefully) cannot be extracted)

But passwords are typically not safe to use as keys for standalone security systems (e.g., encryption systems) that expose data to enable offline password guessing by an attacker. Passphrases are theoretically stronger, and so should make a better choice in these cases. First, they usually are (and always should be) much longer—20 to 30 characters or more is typical—making some kinds of brute force attacks entirely impractical. Second, if well chosen, they will not be found in any phrase or quote dictionary, so such dictionary attacks will be almost impossible. Third, they can be structured to be more easily memorable than passwords without being written down, reducing the risk of hardcopy theft. However, if a passphrase is not protected appropriately by the authenticator and the clear-text passphrase is revealed its use is no better than other passwords. For this reason it is recommended that passphrases not be reused across different or unique sites and services.

In 2012, two Cambridge University researchers analyzed passphrases from the Amazon PayPhrase system and found that a significant percentage are easy to guess due to common cultural references such as movie names and sports teams, losing much of the potential of using long passwords.[6]

When used in cryptography, commonly the password protects a long (machine generated) key, and the key protects the data. The key is so long a brute force attack (directly on the data) is impossible. A key derivation function is used, involving many thousands of iterations (salted & hashed), to slow down password cracking attacks.

Passphrase selection

Typical advice about choosing a passphrase includes suggestions that it should be:[7]

- Long enough to be hard to guess

- Not a famous quotation from literature, holy books, et cetera
- Hard to guess by intuition—even by someone who knows the user well
- Easy to remember and type accurately
- For better security, any easily memorable encoding at the user's own level can be applied.
- Not reused between sites, applications and other different sources.

Example methods

One method to create a strong passphrase is to use dice to select words at random from a long list, a technique often referred to as diceware. While such a collection of words might appear to violate the "not from any dictionary" rule, the security is based entirely on the large number of possible ways to choose from the list of words and not from any secrecy about the words themselves. For example, if there are 7776 words in the list and six words are chosen randomly, then there are $7776^6 = 221073919720733357899776$ combinations, providing about 78 bits of entropy. (The number *7776* was chosen to allow words to be selected by throwing five dice. $7776 = 6^5$) Random word sequences may then be memorized using techniques such as the memory palace.

Another is to choose two phrases, turn one into an acronym, and include it in the second, making the final passphrase. For instance, using two English language typing exercises, we have the following. *The quick brown fox jumps over the lazy dog*, becomes *tqbfjotld*. Including it in, *Now is the time for all good men to come to the aid of their country*, might produce, *Now is the time for all good tqbfjotld to come to the aid of their country* as the passphrase.

There are several points to note here, all relating to why this example passphrase is not a good one.

- It has appeared in public and so should be avoided by everyone.
- It is long (which is a considerable virtue in theory) and requires a good typist as typing errors are much more likely for extended phrases.
- Individuals and organizations serious about cracking computer security have compiled lists of passwords derived in this manner from the most common quotations, song lyrics, and so on.

The PGP Passphrase FAQ[8] suggests a procedure that attempts a better balance between theoretical security and practicality than this example. All procedures for picking a passphrase involve a tradeoff between security and ease of use; security should be at least "adequate" while not "too seriously" annoying users. Both criteria should be evaluated to match particular situations.

Another supplementary approach to frustrating brute-force attacks is to derive the key from the passphrase using a deliberately slow hash function, such as PBKDF2 as described in RFC 2898 .

Windows support

If backward compatibility with Microsoft LAN Manager is not needed, in versions of

Windows NT (including Windows 2000, Windows XP and later), a passphrase can be used as a substitute for a Windows password. If the passphrase is longer than 14 characters, this will also avoid the generation of a *very* weak LM hash.

Unix support

In recent versions of Unix-like operating systems such as Linux, OpenBSD, NetBSD, Solaris and FreeBSD, up to 255-character passphrases can be used.

See also

- Keyfile

- Password-based cryptography

References

1. *Sigmund N. Porter. "A password extension for improved human factors". Computers and Security, 1(1):54-56, January 1982.*

2. *Matt Mahoney. "Refining the Estimated Entropy of English by Shannon Game Simulation" . Florida Institute of Technology. Retrieved March 27, 2008.*

3. *"Electronic Authentication Guideline"* (PDF)*. NIST. Retrieved September 26, 2016.*

4. *Jesper M. Johansson. "The Great Debates: Pass Phrases vs. Passwords. Part*

2 of 3" . Microsoft Corporation. Retrieved March 27, 2008.

5. *Joseph Bonneau, Ekaterina Shutova, Linguistic properties of multi-word passphrases , University of Cambridge*

6. *Godwin, Dan (14 March 2012). "Passphrases only marginally more secure than passwords because of poor choices" . Retrieved 9 December 2014.*

7. *Lundin, Leigh (2013-08-11). "PINs and Passwords, Part 2" . Passwords. Orlando: SleuthSayers.*

8. *Randall T. Williams (1997-01-13). "The Passphrase FAQ" . Retrieved 2006-12-11.*

External links

- Diceware page
- xkcd Password Strength common-viewed explanation of concept

Retrieved from "https://en.wikipedia.org/w/index.php?title=Passphrase&oldid=818378350"

Last edited 8 months ago by Guy M...